[36] J. Shi, and C. Tomasi, “Good Features to Track,” *IEEE International Conference on Computer Vision & Pattern Recognition,* pp. 593–600, 1994.

[37] J.R. Shewchuk, “Delaunay Refinement Mesh Generation,” *Ph.D. Thesis, Carnegie Mellon University,* pp. 593–600, 1997.

[38] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang, “Framework for Performance Evaluation of Face, Text, and Vehicle Detection and Tracking in Video: Data, Metrics, and Protocol,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 31, no. 2, pp. 319–336, 2009.

[25] R. Lienhart, and J. Maydt, “An Extended Set of Haar-Like Features for Rapid Object Detection,” *IEEE International Conference on Image Processing,* vol. 1, pp. 900–903, 2002.

[26] K.M. Lam, and H. Yan, “Locating Head Boundary by Snakes,” *International Symposium on Speech, Image Processing and Neural Networks,* vol. 1, pp. 17–20, 1994.

[27] Taimori, and A. M. Nasrabadi, “An Automatic Method for Human Eye Detection and Tracking Based on Artificial Neural Network and Feature Matching,” *14th Iranian Conference on Biomedical Engineering,* pp. 426–433, 2008 (Printed in Persian).

[28] Taimori, A.R. Behrad, and S. Sabouri, “Automatic Human Face Detection and Tracking in Video Sequences Using a Head Pose Rotation Insensitive Method,” *16th Iranian Conference of Electrical Engineering,* pp. 910–915, 2008 (Printed in Persian).

[29] Y. Wang, and O. Lee, “Active Mesh-A Feature Seeking and Tracking Image Sequence Representation Scheme,” *IEEE Transactions on Image Processing,* vol. 3, no. 5, pp. 610–624, 1994.

[30] D. Molloy, and P.F. Whelan, “Active-Meshes,” *Pattern Recognition Letters,* pp. 1071–1080, 2000.

[31] A.R. Behrad, and S.A. Motamedi, “Moving Target Detection and Tracking Using Edge Features Detection and Matching,” *IEICE Transactions on Information and Systems,* vol. E86–D, no. 12, pp. 2764–2774, 2003.

[32] Jamasbi, S.A. Motamedi, and A.R. Behrad, “Contour Tracking of Targets with Large Aspect Change,” *Journal of Multimedia,* vol. 2, no. 6, pp. 7–14, 2007.

[33] L.S. Shapiro, “Towards a Vision-Based Motion Framework,” *Technical Report, Department of Engineering Science, Oxford University,* 1991.

[34] S. Kim, and I.S. Kweon, “Automatic Model-Based 3D Object Recognition by Combining Feature Matching with Tracking,” *Machine Vision and Applications,* vol. 16, no. 5, pp. 267–272, 2005.

[35] Taimori, A.R. Behrad, and A. Delforouzi, “A New Method for Video Tracking of Flying Targets Based on Energy Functions Minimization,” *16th Iranian Conference of Electrical Engineering,* pp. 329–334, 2008 (Printed in Persian).

Detection Based on the Harmonic Images," 5^{th} *Pacific Rim Conference on Multimedia,* pp. 585–592, 2004.

[14] S. Shan, P. Yang, X. Chen, and W. Gao, "AdaBoost Gabor Fisher Classifier for Face Recognition," *IEEE International Workshop on Analysis and Modeling of Faces and Gestures,* pp. 278–291, 2005.

[15] S. R. Gunn, and M. S. Nixon, "A Dual Active Contour for Head Boundary Extraction," *IEE Colloquium on Image Processing for Biometric Measurement,* pp. 6/1–6/4, 1994.

[16] K-H. An, D-H. Yoo, S-U. Jung, and M-J. Chung, "Robust Multi-View Face Tracking," *International Conference on Intelligent Robots and Systems,* pp. 1905–1910, 2005.

[17] G-Q. Zhao, L. Chen, and G-C. Chen, "A Simple 3D Face Tracking Method Based on Depth Information," *IEEE International Conference on Machine Learning and Cybernetics,* vol. 8, pp. 5022–5027, 2005.

[18] C. Lerdsudwichai, and M. Abdel-Mottaleb, "Algorithm for Multiple Faces Tracking," *International Conference on Multimedia and Expo,* vol. 2, pp. II-777–II-780, 2003.

[19] J. Tu, T. Huang, and H. Tao, "Face as Mouse Through Visual Face Tracking," 2^{nd} *Canadian Conference on Computer and Robot Vision,* pp. 339–346, 2005.

[20] H. Nanda, and K. Fujimura, "Illumination Invariant Head Pose Estimation Using Single Camera," *IEEE Intelligent Vehicles Symposium,* pp. 434–437, 2003.

[21] P. Corcoran, M. C. Jonita, and J. Bacivarov, "Next Generation Face Tracking Technology Using AAM Techniques," *International Symposium on Signals, Circuits and Systems,* vol. 1, pp. 1–4, 2007.

[22] F. Dornaika, and A. D. Sappa, "Evaluation of an Appearance-Based 3D Face Tracker Using Dense 3D Date," *Machine Vision and Applications,* 2007.

[23] P. Viola, and M.J. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features," *IEEE International Conference on Computer Vision & Pattern Recognition,* vol. 1, pp. I-511–I-518, 2001.

[24] P. Viola, and M.J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137-154, 2004.

[3] B. Yip, "Face and Eye Rectification in Video Conference Using Artificial Neural Network," *IEEE International Conference on Multimedia and Expo,* pp. 690–693, 2005.

[4] C-Y. Tsai, and K-T Song, "Face Tracking Interaction Control of a Nonholonomic Mobile Robot," *IEEE International Conference on Intelligent Robots and Systems,* pp. 3319–3324, 2006.

[5] C-C. Chang, and H. Aghajan, "Linear Dynamic Data Fusion Techniques for Face Orientation Estimation in Smart Camera Network," *IEEE International Conference on Distributed Smart Cameras,* pp. 44–51, 2007.

[6] C. Kozasa, H. Fukutake, H. Notsu, Y. Okada, and K. Niijima, "Facial Animation Using Emotional Model," *International Conference on Computer Graphics, Imaging and Visualization,* pp. 428–433, 2006.

[7] M. J. Er, W. Chen, and S. Wu, "High-Speed Face Recognition Based on Discrete Cosine Transform and RBF Neural Networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 679–691, 2005.

[8] L. Ma, T. Tan, Y. Wang, and D. Zhang, "Personal Identification Based on Iris Texture Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 25, no. 12, pp. 1519–1533, 2003.

[9] F. Morganti, M. L. Rusconi, A. Cantagallo, E. Mondin, and G. Riva, "A Context-Based Interactive Evaluation of Neglect Syndrome in Virtual Reality," *Virtual Rehabilitation,* pp. 169–174, 2007.

[10] S. Mitra, and T. Acharya, "Gesture Recognition: A Survey," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 37, no. 3, pp. 311–324, 2007.

[11] E. Y. Kim, and S. H. Park, "Computer Interface Using Eye Tracking for Handicapped People," *7th International Conference on Intelligent Data Engineering and Automated Learning,* pp. 562–569, 2006.

[12] J-B. Kim, S-W. Jung, and H-J. Kim, "Face Detection by Integrating Multiresolution-Based Watershed and a Skin-Color Model," *15th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems,* pp. 715–724, 2002.

[13] J. Chen, Y. Li, L. Qing, B. Yin, and W. Gao, "Face Samples Re-Lighting for

In Fig. 14, seven points are selected manually on the mouth to demonstrate the effects of internal and external energies for tracking the human mouth. The results show that the structure energy function is efficient in constructing the mouth mesh structure. At video frame 35, the effect of structure energy is obvious. The internal energy is able to preserve the constructed meshes by Delaunay triangulation and the external energy can track non-rigid motions of the mouth accurately.

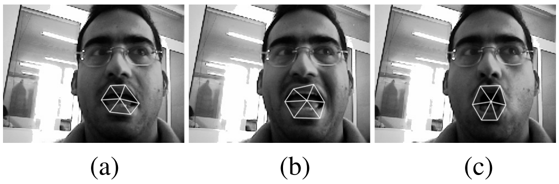

Fig. 14. Mouth tracking; (a) frame 13, (b) frame 24, and (c) frame 35.

Table 3. results of the proposed face tracking algorithm for six different videos

Input Test Video	SE	ME	Variance	RMSTE	RTE
Video A	13.18	15.74	1.14	15.78	2.56
Video B	12.56	10.08	0.82	10.11	2.48
Video C	8.94	6.05	3.65	6.33	2.89
Video D	6.34	7.95	0.91	8.01	1.61
Video E	11.87	13.12	2.46	13.21	1.25
Video F	10.20	14.96	4.69	15.11	4.76

6- Conclusion

In this paper, a novel algorithm is proposed for face detection and tracking. The proposed face detection and tracking modules are based on the cascaded classifiers and a combination of feature matching and deformable mesh models, respectively. The proposed face detection method is able to locate rotated faces in images.

Since the human face is a semi-rigid object, we defined a new set of energy functions to improve the accuracy and stability of the face region tracking algorithm. The proposed method has the following advantages:

- It is insensitive to head pose rotations.
- works well on the low resolution video sequences (i.e., at resolution of 320×240 pixels).
- It is inexpensive and needs only a low resolution camera (like a Web camera) for tracking the face.

The proposed method was tested with Gray-level video frames and experimental results shown robustness and computational efficiency of the proposed algorithms. The proposed method is appropriate for accurate and continuous face tracking applications like camera mouse and monitoring driver fatigues.

For future researches, we suggest generalization of the proposed algorithm to RGB video sequences and energy functions definition based on the color information of human faces. In our future works, we also aim to consider partial occlusion in mesh based approaches.

Table 4. Comparision two face tracking algorithms

Method	SE	ME	RTE
Our Method	10.51	11.31	0.8
Method of [30]	10.51	19.26	8.75

7- References

[1] Y. Shin, and E. Y. Kim, “Welfare Interface Using Multiple Facial Features Tracking,” *19th Australian Joint Conference on Artificial Intelligence,* pp. 453–462, 2006.

[2] Y. Zhu, and K. Fujimura, “Driver Face Tracking Using Gaussian Mixture Model,” *IEEE Intelligent Vehicles Symposium,* pp. 587–592, 2003.

as mouth and eye. Otherwise, if $\zeta = 0.5$, the internal and external energies have same effects in the tracking algorithm. Therefore, we should compromise between 2-D and 3-D motions and/or rigid and non-rigid motions to attain the best performance by selecting the best parameters adaptively.

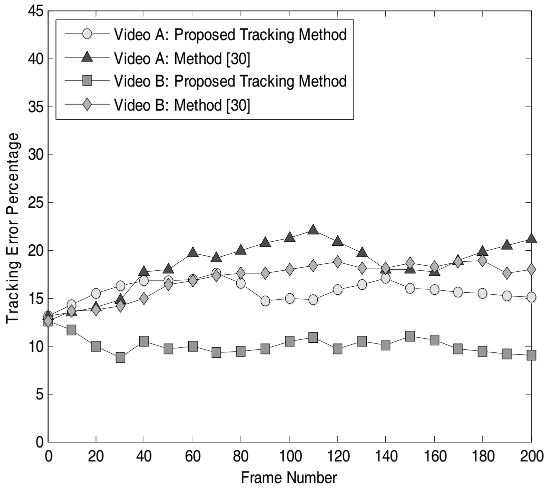

Fig. 9. TEP in terms of frame number for videos A and B in the proposed method and the method of Molloy and Whelan [30].

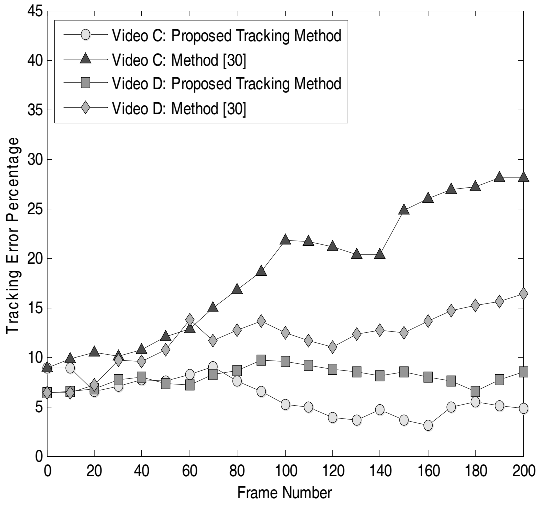

Fig. 10. TEP in terms of frame number for videos C and D in the proposed method and the method of Molloy and Whelan [30].

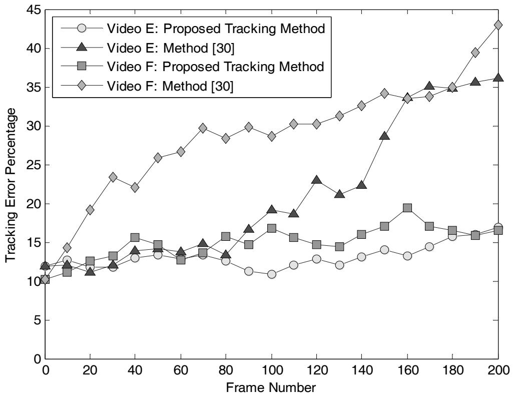

Fig. 11. TEP in terms of frame number for videos E and F in the proposed method and the method of Molloy and Whelan [30].

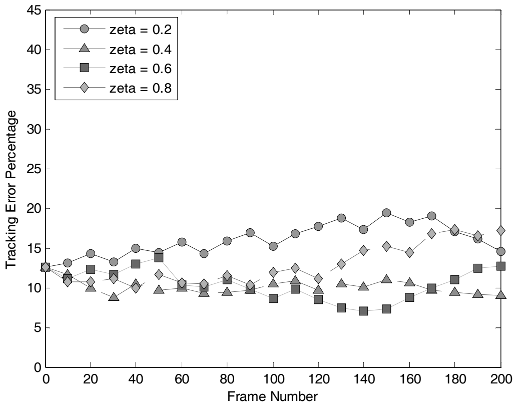

Fig. 12. Effects of the regularization parameter on video B.

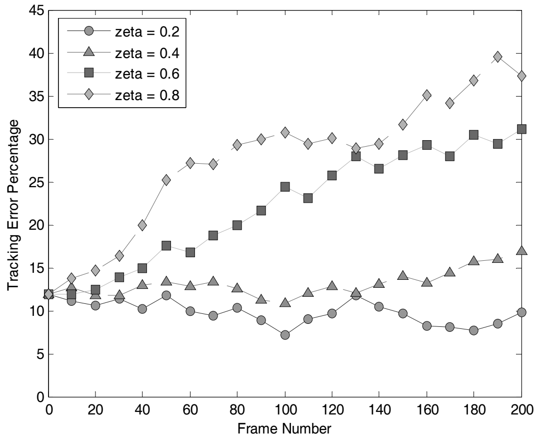

Fig. 13. Effect of the regularization parameter on video E.

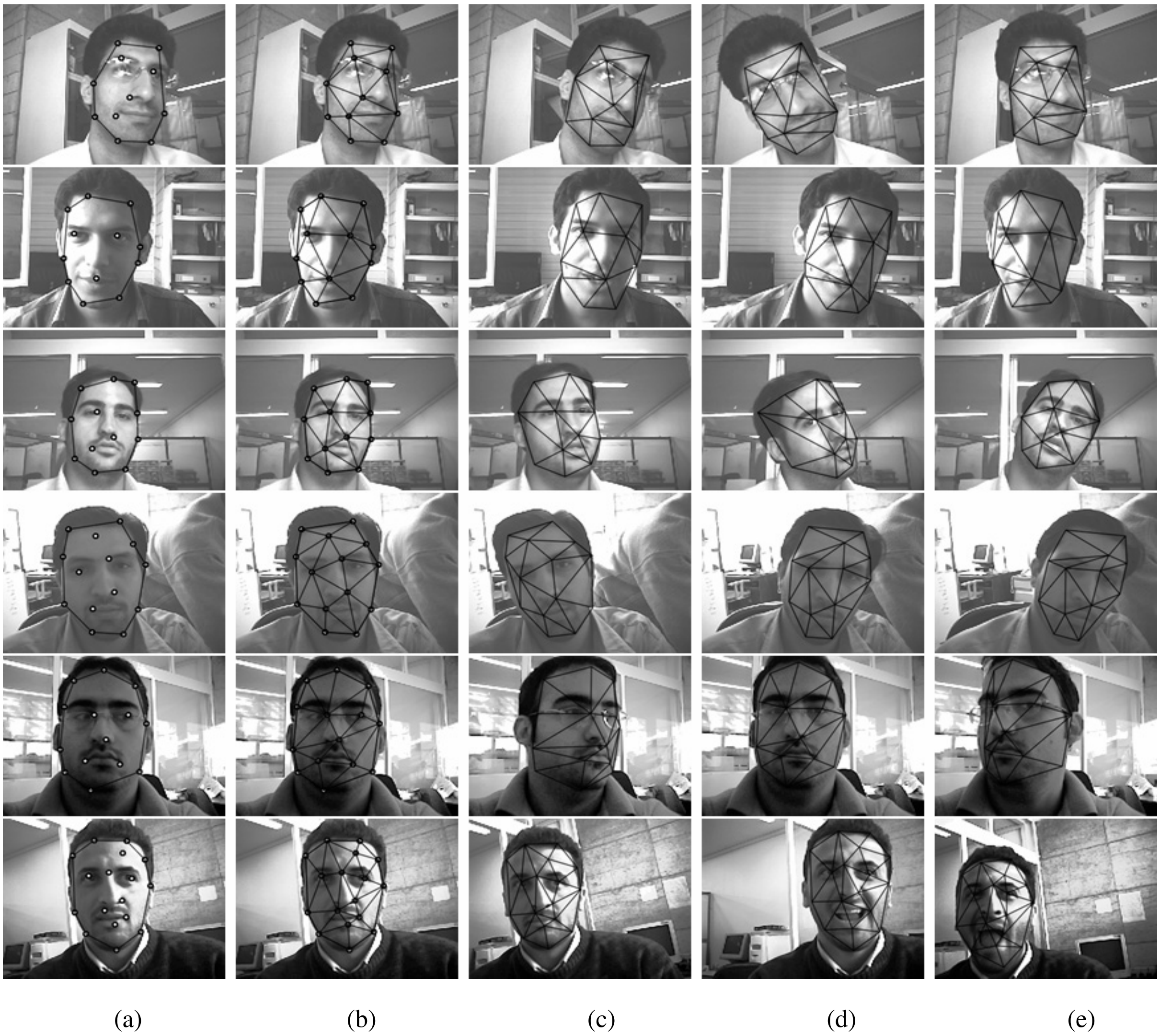

Fig. 8. Results of the proposed face tracking algorithm for six different videos (A, B, C, D, E, and F); (a) face features and contour extraction (b) face mesh generation (c) ,(d) ,(e) results of the tracking algorithm at different video frames; the first row: frames 70, 149, and 200; the second row: frames 40, 85, and 190; the third row: frames 112, 132, and 198; the fourth row: frames 45, 145, and 192; the fifth row: frames 90, 128, and 158; the sixth row: frames 58, 149, and 239.

5-3 Regularization parameter

As mentioned previously, we defined regularization parameter, ζ, to balance the influence of internal and external energies. Figs. 12 and 13 describe the effects of changing the regularization parameter for videos B and E, respectively. For video B, $\zeta = 0.4$ yields better results. However, as shown in Fig. 8, the person in video E has 3-D head motion; so $\zeta = 0.2$ yields the least error. In fact, this parameter regularizes the terms of internal and external energies; in other words, if $\zeta > 0.5$, then the internal energy is the dominant term and makes the mesh suitable for rigid and 2-D motions like head translations. If $\zeta < 0.5$, then the external energy is more effective than the internal energy; therefore mesh can model non-rigid and 3-D motions of the face and facial features such

Table 2. results of the face detection algorithm

Input Image	Test Data	False Positive	False Negative	FDR
Non-Face	46	3	-	93.47%
Face	55	-	4	92.72%

5-2 The Second Module Results: FaceTracking

Fig. 8 shows face feature extraction, face boundary detection and face tracking results for six different videos (A, B, C, D, E, and F) in different situations like fixed and moving camera, put on glasses, head rotation, partial illumination variations, aspect changes and face and facial features movements.

In order to evaluate the accuracy of the proposed tracking method and compare the results with that of the another method, we have defined a criterion to measure tracking error in video sequences for the human face region which is similar to performance evaluation definitions for face tracking in [38]. For this purpose, the tracking error percentage (TEP) at frame t is defined as:

$$TEP(t) = 100 \times \left(1 - \frac{S(F \cap M)}{S(F \cup M)} \right), \qquad (23)$$

where F and M are the face and mesh contours, respectively, and $S(C)$ represents the surface of the closed contour of C. For calculating the TEP, the face contour (i.e., face "ground truth") is manually determined which contains the human face boundary except the hair and ears regions. results of the proposed face tracking method at each video frame compares with the relative face "ground truth."

For comparing the results of the proposed tracking algorithm with that of the another method, we also implemented the method of Molloy and Whelan [30] which uses a mesh based tracking algorithm. Fig. 9 shows the TEP versus the frame number of videos A and B for the proposed method and the method of [30]. Figs. 10 and 11 depict the TEP values in terms of frame number for videos C, D and E, F, respectively.

In order to compare the average tracking error of different methods, different criteria may be used such as mean absolute tracking error (MATE) and root mean squared tracking error (RMSTE). However, MATE and RMSTE do not consider error at the first frame of the video which is due to the mesh and contour surface difference. Therefore, we defined error at the first frame as static error (SE) and the absolute value of the difference between MATE and SE as the true tracking error (RTE).

Table 3 shows the criterion values for the proposed face tracking algorithm for different videos in Fig. 8.

Table 4 compares average results of the proposed method with the method of [30] with a same SE. As shown in this table, results of the proposed algorithm are superior to the results of [30]. This table shows that the RTE error for the proposed algorithm is about 11 times less than that of [30]. This performance has been obtained by combining a feature matching algorithm and the proposed deformable mesh model.

The peak RTE in the proposed method is limited to low percents; however, in the method of Viola and Jones [23, 24] which is, in fact, a face detector in video sequences, it is 100% for rotated human faces. Therefore, the face is lost in such situations. This ability of the suggested method is suitable for human-machine interfacing systems which require accurate tracking.

meshes. The Greedy algorithm for minimization of the mesh energy and tracking the face includes the following steps:

1) Select one of the unselected vertices of mesh, e.g., vertex i.

2) Fix the location of other vertices and move the selected vertex in the window centered on the initial location of the selected vertex.

3) Calculate the total mesh energy for all the possible locations in step 2.

4) Move the vertex to the point with minimum energy, $\min_{\mathbf{v}_i}\{E_{\text{Mesh}}\}$.

5) Repeat steps 1 to 4 to reach the maximum iteration $(Iter_{\max})$.

6) Update mesh elements and parameters (ML,MV).

5- Experimental Results

The proposed FDAT algorithm implemented using the Microsoft Visual C++ 8.0 compiler and tested on an Intel Pentium-IV 3.2GHz personal computer (PC) to detect and track human faces using a PC camera with frame rate of 20 frames/s and frame size of 320×240 pixels. in an indoor environment. The proposed method was tested on several video samples in different situations like cluttered background, put on glasses, static and moving camera. Head motions may introduce 2-D and 3-D rotations and translations. execution time for processing a video frame in the proposed algorithm with the mentioned PC was about 60 ms. resulting in a frame rate of 16 frames/s for continuous tracking of human faces; whereas, the frame rate of the proposed method given in [24] was about 15 frames/s.

We tried to use the optimal parameters for the proposed face detection and tracking algorithm. The value of these parameters in our experiments was as: $m=12$, $p=21$, $\omega=7$, $k=8$, $\varphi=0.65$, $Iter_{\max}=12$, $\alpha=1\&0$ (based on illumination change in video sequences), $\beta=0\&1$ (based on illumination change in video sequences), $\gamma=0.9$, $\eta=1$ and $\zeta=0.4$. The optimal parameters have been obtained by trial and error to attain an acceptable performance.

5-1 The First Module Results: Face Detection

Fig. 7 shows the results of face detection and head pose estimation algorithms at the first frame for different images. The proposed face detection approach is robust to rotations of human head up to $\theta \cong \pm 35^\circ$.

Table 2 lists results of the cascaded classifiers for 101 images including face and non-face images. As shown in this table, the average face detection rate (FDR) was approximately 93.1% for the tested dataset. The rate of false positive (false acceptance) and false negative (false rejection) for 46 non-face samples and 55 face samples was 3 and 4, respectively. In order to measure the performance of the face detection algorithm, face "ground truth" has defined which comprises facial features like eyes, eyebrows, nose and mouth. Thus, the results of the detection method compares based on this criterion.

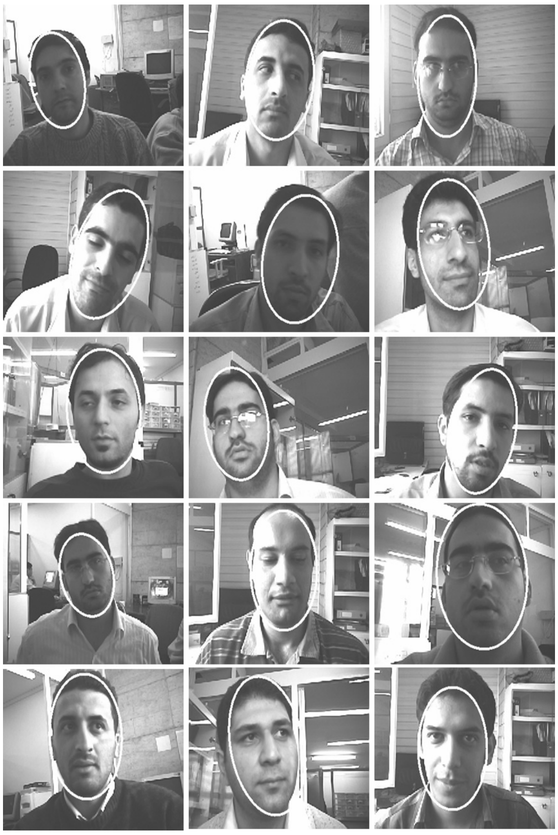

Fig. 7. Face detection and head pose estimation results at the first video frame.

$$E_{\text{IC}}(i)=\frac{\max\limits_{k-\ \text{nib}}\{R\}-R(i)}{\max\limits_{k-\ \text{nib}}\{R\}}, \tag{18}$$

where R is the cornerness function of the feature extraction algorithm as:

$$R(i)=\left|\lambda_1\lambda_2-\varphi(\lambda_1+\lambda_2)\right|, \tag{19}$$

where φ is a constant factor, and λ_1 and λ_2 are two eigenvalues of the following 2×2 matrix:

$$Z(i)=\begin{pmatrix}\left\langle f_x^2(y_i,x_i,t)\right\rangle & \left\langle f_x(y_i,x_i,t)f_y(y_i,x_i,t)\right\rangle \\ \left\langle f_x(y_i,x_i,t)f_y(y_i,x_i,t)\right\rangle & \left\langle f_y^2(y_i,x_i,t)\right\rangle\end{pmatrix}. \tag{20}$$

4-3 Selection of the Face Boundary Features

In order to determine external nodes of the face mesh for calculating the edge energy, we approximate the face contour with a convex hull where the external nodes are vertices of the convex hull. Fig. 6 illustrates the suggested algorithm for finding the convex hull. The proposed algorithm comprises the following steps:

1) Calculate the center of mass, $P_0=(\hat{y}_0,\hat{x}_0)$, for mesh nodes.

2) Find the farthest node to the center of mass. This node, $P_1=(\hat{y}_1,\hat{x}_1)$, is specified as the first convex hull vertex.

3) Determine the infinite line, normal to the line segment from the obtained convex hull vertex to the center of mass, P_0P_1. The equation of the infinite line is given by:

$$y-m_2x+m_2\hat{x}_1-\hat{y}_1=0, \tag{21}$$

where $m_2=-1/m_1$ is the slope of the infinite line and $m_1=(y-\hat{y}_0)/(x-\hat{x}_0)$ is the slope of the line segment P_0P_1.

4) Rotate the infinite line clockwise around the obtained convex hull vertex gradually until it intersects the mesh in a new node and add the obtained node to convex hull vertices.

5) Repeat steps 3 and 4 for the newly obtained nodes until reaching the first obtained node of the convex hull.

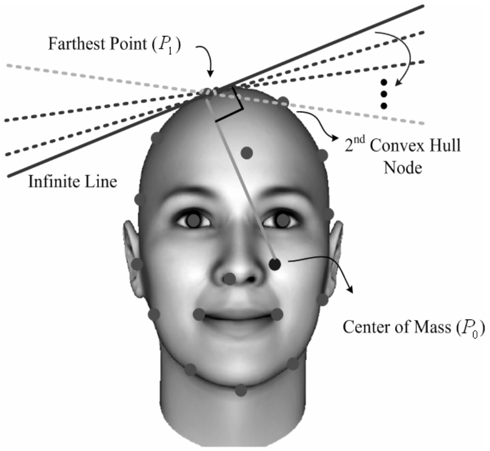

Fig. 6. The proposed scheme for finding convex hull vertices.

4-4 Energy Combinations

The total energy of the mesh in the discrete domain is a weighted sum of the vertex energies where mesh vertices are face features at the first video frame or their matches at subsequent frames. We define the mesh energy as the sum of the internal and external energies as follows:

$$E_{\text{Mesh}}=\sum_{i=0}^{N-1}\left\{\zeta E_{\text{Internal}}(i)+[1-\zeta]E_{\text{External}}(i)\right\}, \tag{22}$$

where $E_{\text{Internal}}(i)$ and $E_{\text{External}}(i)$ represent the internal and external energies of vertex i, respectively, N is the total number of mesh nodes, and ζ is a user defined coefficient in the range $0\le\zeta\le 1$. In fact, ζ is a regularization parameter which balances the effect of internal and external energies for the mesh nodes.

4-5 The Optimization Approach

We have used the Greedy algorithm [15] to minimize the calculated energies in deformable of vertex i, respectively, $E^{*}_{\text{External}}(i)$ is the sum of matching and interest points energies which lies in the range of $[0,4]$, and α, β, γ, and η are user defined external energy coefficients in the range $[0,1]$. In fact, equation (9) is normalized to map external energy from $[0,4]$ to $[0,1]$.

matching energy tries to attract mesh vertices towards the most similar points. This energy is based on short time interval between the two consecutive frames. In other words, matching energy tries to minimize the similarity error between the two features at frames t and $t-1$. We define this energy as:

$$E_{\text{Matching}}(i) = \alpha E_{\text{Correlation}}(i) + \beta E_{\text{SSDs}}(i), \qquad (11)$$

where $E_{\text{Correlation}}(i)$ and $E_{\text{SSDs}}(i)$ are the correlation and SSDs energies in gray level video sequences, respectively. The correlation method is not sensitive to constant intensity variations; however SSDs considers it. The combination of these energies has the advantage of using the benefits of both methods. For videos with illumination flactuations, we use the correlation energy by setting $\alpha = 1$ and $\beta = 0$, otherwise SSDs energy is utilized by setting $\alpha = 0$ and $\beta = 1$. The normalized correlation energy for each vertex in the mesh is calculated as:

$$E_{\text{Correlation}}(i) = \frac{1}{2}\{1 - r_{\text{NCC}}(\mathbf{v}_i)\}, \qquad (12)$$

where $r_{\text{NCC}}(\mathbf{v}_i)$ is the normalized cross-correlation (NCC) between the two selected features at frames t and $t-1$ which is calculated using image pixel intensities in the $\omega \times \omega$ windows centered on feature points.

The normalized SSDs energy is given by:

$$E_{\text{SSDs}}(i) = \frac{r_{\text{SSDs}}(\mathbf{v}_i)}{\max\limits_{k- \ \text{nib}} \{r_{\text{SSDs}}(\mathbf{v}_i)\}}, \qquad (13)$$

where $r_{\text{SSDs}}(\mathbf{v}_i)$ is the weighted SSDs between the two selected features at frames t and $t-1$.

The second term of $E^{*}_{\text{External}}(i)$ is the interest point energy. As mentioned before, our goal in defining this energy is to move the internal nodes towards the corner points and the external nodes towards the edge and corner points. This energy is defined as:

$$E_{\text{IPs}}(i) = \gamma E_{\text{IE}}(i) + \eta E_{\text{IC}}(i), \qquad (14)$$

where $E_{\text{IE}}(i)$ and $E_{\text{IC}}(i)$ are the image edge and corner energies, respectively. The edge energy is calculated only for external or contour points of the face, whereas the corner energy is obtained for all the face features. The normalized edge energy function is given by:

$$E_{\text{IE}}(i) = \Psi(i)\left(\frac{\max\limits_{k- \ \text{nib}} \{G\} - G(i)}{\max\limits_{k- \ \text{nib}} \{G\} - \min\limits_{k- \ \text{nib}} \{G\}}\right), \qquad (15)$$

where G is the squared magnitude of the image gradient at vertex location $\mathbf{v}_i$ as:

$$G(i) = \left\langle f_x^2(y_i, x_i, t)\right\rangle + \left\langle f_y^2(y_i, x_i, t)\right\rangle, \qquad (16)$$

Where $\langle \cdot \rangle$ represent the Gaussian smoothed version of the image, and $\Psi(i)$ is an adaptive coefficient defined as:

$$\Psi(i) = \begin{cases} 1 & \text{if } \mathbf{v}_i \text{ is in the face boundary} \\ 0 & \text{otherwise.} \end{cases} \qquad (17)$$

In order to obtain $\Psi(i)$, it is necessary to extract nodes in the boundary of the face. We extract a convex hull enclosing the face to do so, and its algorithm will be described in Sub-section 4-3.

Since the face features at the first frame are extracted using the KLT approach, it is assumed that the matched points at the consecutive frames also have the properties of features at the first one. Based on this idea, we have defined the corner energy function based on KLT parameters. Thus, the normalized corner energy function is:

neighbors in different video sequences. Considering both the rigid and non-rigid motion of the face, we have defined this energy to preserve only local shape of the mesh not the global shape. The minimization of this energy makes the mesh having a similar local shape variations and motion of the previous frames. This also means to use the motion information of the previous frames to find the true match points at the current frame. The idea of the suggested structure energy is formulated by:

$$E_{\text{Structure}}(i)=\frac{\left\|\bar{d}_i(t)-\left|\bar{h}_i(t)-\bar{h}_i(t-1)\right\|\right\|^2}{\max\limits_{k-\ \text{nib}}\left\{\left\|\bar{d}_i(t)-\left|\bar{h}_i(t)-\bar{h}_i(t-1)\right\|\right\|^2\right\}},\quad (6)$$

where $E_{\text{Structure}}(i)$ is the structure energy of node i, the denominator has been used to normalize the structure energy between $[0,1]$, $k-\text{nib}$ means the k nearest neighbors for each mesh node, $\bar{d}_i(t)$ is the average displacement of node $\mathbf{v}_i$ at frame t relative to its node neighbors at frame $t-1$, and $\bar{h}_i(t)$ and $\bar{h}_i(t-1)$ are the average Euclidean distances of node $\mathbf{v}_i$ from its neighboring nodes at frames t and $t-1$, respectively. $\bar{d}_i(t)$ and $\bar{h}_i(t)$ are calculated using the following equations:

$$\bar{d}_i(t)=\frac{\sum\limits_{u=0}^{n_i-1}\left\|\mathbf{v}_i(t)-\mathrm{N}_{\mathbf{v}_{iu}}(t-1)\right\|}{n_i},\quad (7)$$

$$\bar{h}_i(t)=\frac{\sum\limits_{u=0}^{n_i-1}\left\|\mathbf{v}_i(t)-\mathrm{N}_{\mathbf{v}_{iu}}(t)\right\|}{n_i},\quad (8)$$

where n_i is the number of nodes in the neighborhood of node $\mathbf{v}_i$, $\mathrm{N}_{\mathbf{v}_{iu}}$ is u^{th} node in the neighborhood of $\mathbf{v}_i$, and $\|\cdot\|$ represents Euclidian distance. It is important to note that equation (7) is updated after each iteration in the minimization algorithm. Fig. 5 illustrates different parameters of the internal energy for a single node.

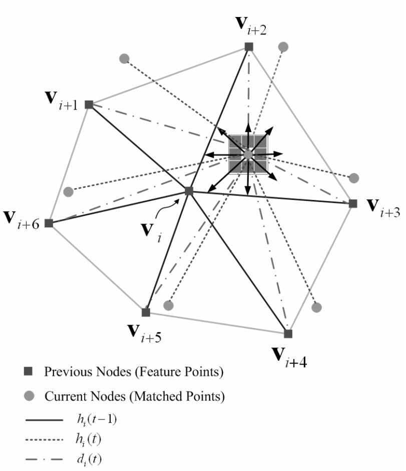

Fig. 5. Different parameters of internal energy for a single node.

4-2 The External Energy Function

external energy utilizes image information such as intensity and texture to locate the true matching points of the mesh nodes. We have defined the external energy based on the following intuitions:

- Considering the short interval between the frames, nodes are mostly matched with similar points at next frames.
- Since nodes are the interest points (corners) of the first frame, the nodes are probably matched with corners or interest points at subsequent frames.
- Face rotation and/or aspect change of the camera make the new parts of face appear or some parts of the face disappear. In order to cope with these problems, external nodes of mesh should be attracted to external boundaries or edges of the face.

Based on the above intuitions, we have defined the normalized external energy as follow:

$$E_{\text{External}}(i)=\frac{E^{*}_{\text{External}}(i)}{\alpha+\beta+\gamma+\eta},\ \text{such that}\quad (9)$$

$$E^{*}_{\text{External}}(i)=E_{\text{Matching}}(i)+E_{\text{IPs}}(i),\quad (10)$$

where $E_{\text{Matching}}(i)$ and $E_{\text{IPs}}(i)$ are the matching energy and the interest point energy

$$\left(\frac{x'-x_0}{\mathrm{a}}\right)^2+\left(\frac{y'-y_0}{\mathrm{b}}\right)^2-1\leq 0. \qquad (4)$$

When the features inside the face regions are extracted, they are used to constitute the face mesh. Fig. 4 shows the mesh initialization using features.

4- The Face Tracking Module

Face features are tracked by minimization of the mesh energy. We have used non-overlapping triangular meshes which are generated by the Delaunay triangulation algorithm to model the human face [37]. Table 1 shows different elements and parameters of the proposed mesh model. These parameters are used to formulate mesh energies. In next sub-sections, we will discuss the proposed internal and external energies.

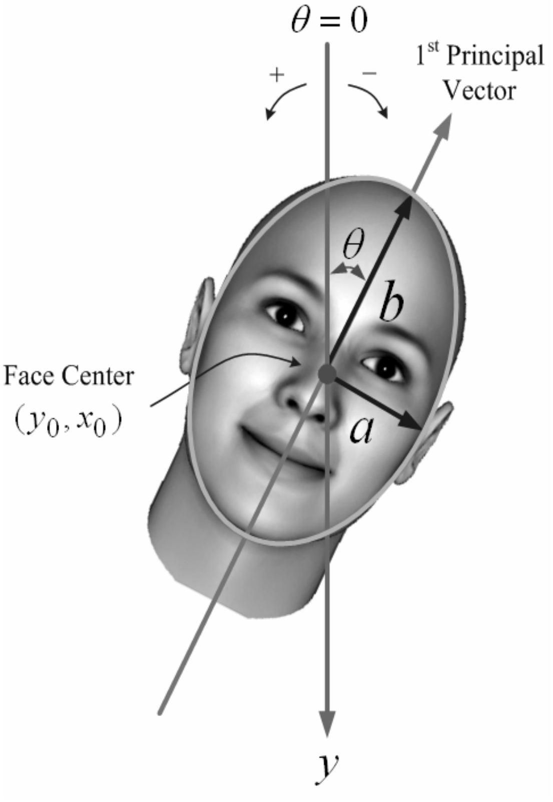

Fig. 3. The scheme of 2-D head pose rotation estimation

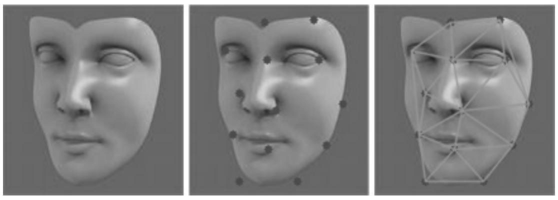
(a) (b) (c)

Fig. 4. The mesh initialization routine; (a) segmented image, (b) feature extraction, and (c) face modeling using triangular meshes.

Table 1. Mesh elements and parameters

Descriptions	Parameters
Mesh lines	$\mathrm{ML}=\{l_0,l_1,\ldots,l_{L-1}\}$
Mesh vertices (nodes)	$\mathrm{MV}=\{\mathbf{v}_0,\mathbf{v}_1,\ldots,\mathbf{v}_{N-1}\}$
Location of the i^{th} vertex	$\mathbf{v}_i=(y_i \quad x_i)^{\mathrm{T}}$
The j^{th} line	$l_j=(\mathbf{v}_{\text{org}} \quad \mathbf{v}_{\text{dst}})^{\mathrm{T}}$
Vicinities of the i^{th} vertex	$\mathrm{N}_{\mathbf{v}_i}=\{\mathrm{N}_{\mathbf{v}_{i0}},\mathrm{N}_{\mathbf{v}_{i1}},\ldots,\mathrm{N}_{\mathbf{v}_{in-1}}\}$

4-1 Internal Energy Function

The internal energy is related to the structure and geometric shape of the mesh; whereas the external energy pertains to image features and tries to attract mesh nodes toward image features such as the face boundaries and the facial features. For a single vertex in triangular meshes, the internal energy is given by:

$$E_{\text{Internal}}(i)=E_{\text{Structure}}(i), \qquad (5)$$

where $E_{\text{Structure}}(i)$ is the structure energy of vertex i. The structure energy tries to retain the initial shape of the generated mesh by the Delaunay triangulation based on the physical shape of the object. Each node of the mesh also absorbs other neighbor nodes and/or repels them. This energy takes into account the geometric shape variations between the two consecutive video frames. We have defined this energy to equalize the distance between each vertex and its

A strong classifier at each stage of the cascaded classifiers consists of p linear weak classifiers [24]. After training the cascaded classifiers, the input image is searched using the multi-scale search window to identify the facial region. It is possible to use a set of basic Haar-like features [23] as well as extended Harr-like features [25] as input to the classifier. In order to produce a binary image, we discriminate each pixel of the first frame into two classes: the face class and the non-face class based on the value of the Haar-like wavelet features. When pixels of the face class are identified, we estimate the face center, (y_0, x_0). Then, we approximate the face area with an ellipse in the image.

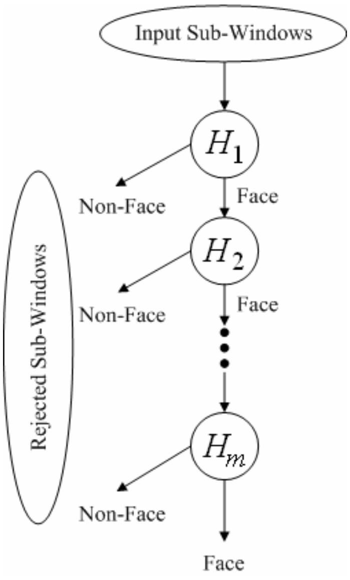

Fig. 2. Schematic of cascaded classifiers

3-1 Head Pose Rotation Estimation

We have proposed an algorithm that is able to localize the face region in image using a rotated ellipse according to the head rotation angle, θ. For estimating the face region with rotated ellipse, it is necessary to obtain head pose orientation. We calculate the rotation angle of the human face relative to vertical axis, y, by applying PCA to the output binary image of the cascaded classifiers. For this purpose, we organize observation matrices, X, as follow:

$$X = \begin{pmatrix} y_1 & y_2 & \cdots & y_d \\ x_1 & x_2 & \cdots & x_d \end{pmatrix}^{\mathrm{T}}, \quad (1)$$

where (y, x) are coordinates of the face pixels and d is the number of face pixels in the binary image. Then, we apply PCA to identify the principal component variances and the related directions to compute the rotation angle. Fig. 3 illustrates our scheme to estimate the head pose orientation. After finding the first principal vector, the face region in image is estimated by a rotated ellipse with the following algebraic equation:

$$\left(\frac{x' - x_0}{\mathrm{a}}\right)^2 + \left(\frac{y' - y_0}{\mathrm{b}}\right)^2 - 1 = 0, \quad (2)$$

where a and b are semi-minor axis and semi-major axis centered at point (y_0, x_0), respectively, and x' and y' are rotation equations in terms of θ as follows:

$$\begin{cases} x' = x\cos(\theta) + y\sin(\theta) \\ y' = y\cos(\theta) - x\sin(\theta), \end{cases} \quad (3)$$

a and b ($a < b$) are determined based on the principal component variances for each sample image. It is important to note that the face detection algorithms are applied to the first frame only; then, the face region is tracked at subsequent frames by the proposed mesh based tracking algorithm.

3-2 Feature Extraction

We extract the face features using the KLT feature extractor which detects face corners or interest points in the image. These feature points are roughly robust to noise and are suitable for face features tracking. We use only interest points which are inside the face region. In other words, we utilize only points which their spatial coordinates in image plane satisfy the following inequality:

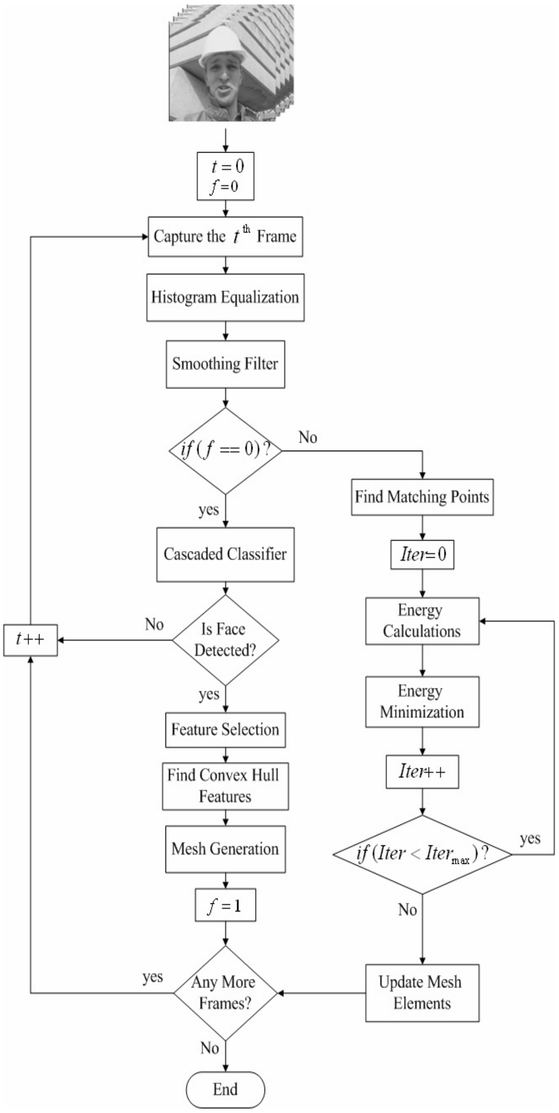

Fig. 1. Flow chart of the proposed FDAT algorithm

When the face is detected, the second module is fired which is the face tracking module. In order to track the face, matching points of the features are found using the improved Lucas-Kanade (LK) feature matching algorithm [27] which utilizes image pyramids for calculation of match points. The method calculates match point of each feature individually with sub-pixel accuracy. However, the match points of some features may not be calculated precisely and are considered as erroneous match. Thus, the match points of the previous step are not considered as the true match points. In order to find the true match points, mesh energy is defined based on the location of feature points, their matches and other attributes of the generated mesh and the face image itself. tracking of face features is performed by minimization of the mesh energy which considers motion information of the nearby features to remove erroneous matches and enhance the accuracy. mesh energy is a combination of internal and external energies [28]. To achieve high accuracy, we have defined different energy functions for mesh, including, matching, interest points (IPs), correlation, sum of squared differences (SSDs), image edge (IE), and image corner (IC) energies. These energies are calculated for match points as well as the predefined neighborhoods around them. mesh energy is the weighted sum mation of the internal and external energies. By minimization of mesh energy at each frame, the location of face features and the related mesh are calculated. In order to track the human head and face at subsequent frames, this routine will resume to algorithm reach the last video frame.

3- The Face Detection Module

We implemented a modified version of cascaded classifiers based on the AdaBoost learning algorithm and the Haar-like wavelet features [23-25] that is able to adaptively estimate the human face region using an ellipse. In this method, multiple classifiers are trained by approximately 4000 image patterns. These patterns consist of both negative and positive samples of size 20×20 pixels. Negative samples are taken from arbitrary images like random matrices, natural pictures, and indoor and outdoor environments. Positive samples contain face images of different cases with different views. The structure of the AdaBoost cascaded classifiers utilized are shown in Fig. 2 which consist of m strong classifiers.

also have some difficulties in tracking the features. In case of rigid motions, motion models such as the planar affine transformation [31, 32] can be employed to estimate the global motion of the object and to reduce tracking errors and reject erroneous matches. However, face motion contains both rigid motions like head motion and non-rigid motions such as the lip and eye motions; therefore the global motion of the face features may not be approximated using a motion model. Hence, face features are mostly tracked independently which makes match points not to be found accurately.

1-2 Our Contribution

In this paper, we propose a novel method for fully automatic human face detection and tracking which can address some of the above mentioned problems in FDAT. The proposed algorithm consists of two modules: face detection and face tracking. The proposed face detection algorithm is based on the AdaBoost cascaded classifier which is robust to head pose rotations. In the tracking module, we have used a technique based on a combination of feature matching algorithms and deformable mesh models to track the human face accurately. In the proposed method, merits of the feature matching algorithm are employed in the mesh model to cope with its problems. This technique allows using the motion information of the nearby features in tracking a specified feature together with the non-rigid motion capability. These are some innovations of this researeh:

1) An adaptive version of AdaBoost cascaded classifiers is proposed to detect rotated human heads in images and video sequences.

2) The proposed tracking algorithm is based on a combination of the feature based and the mesh based tracking algorithms.

3) A new energy function is defined as the internal energy of a mesh.

4) A new set of energy functions are defined as the external energy of a mesh.

The proposed 2-D convex hull detection technique is based on our previous work [35].

The rest of the paper is organized as follows: Section 2 summarizes the outline of the proposed FDAT method. Section 3 explains the modified version of the face detection technique in which the algorithm for finding the head pose rotation as well as the feature extraction approach is discussed. In Section 4, we discuss the face tracking module. This section describes different algorithms such as a new energy function definition for deformable meshes, face boundary detection, energy combinations and energy minimization approaches. Experimental results appear in Section 5, and finally, we conclude the paper in Section 6.

2- Outline of the Proposed Method

As mentioned previously, our proposed FDAT scheme consists of two modules: face detection at the first frame and tracking of the detected face at subsequent video frames. Fig. 1 shows the flow chart of the method. In the detection module, the video frames are captured from video image buffers to detect the human face. In order to detect and track the face, we apply a pre-processing including histogram equalization and smoothing filtering to enhance the frame quality and reduce sensitivity to video noises. After the pre-processing algorithm, the face region is detected using a modified version of Adaboost cascaded classifiers. Then, face features (interest points) are extracted in the face region using the Kanade-Lucas-Tomasi (KLT) feature extractor technique [36]. We used interest points as mesh vertices to generate an unstructured mesh over the face region by the Delaunay triangulation algorithm [37].

proposed an automatic method for face detection in video conferencing that considers head and shoulder views of human. This method segments the input image using the watershed algorithm and the face region is identified by skin-color Gaussian model from the segmented regions. Chen et al. [13] used support vector machine (SVM) to detect the face in different environment illuminations. Shan et al. [14] utilized the AdaBoost Gabor- Fisher classifier for face recognition in static images. Gunn and Nixon [15] used a dual deformable contour to segment human head boundary in low texture images. An et.al. [16] utilized the AdaBoost learning algorithm to train rectangle features into the human face. Then, the face tracking is performed by tracking these features using a linear Kalman filter. Zhao et.al. [17] proposed a 3-D face tracking method based on stereo vision. In order to achieve 3-D tracking, this method extracts and matches 2-D facial features to calculate depth information. Lerdsudwichai and Abdel-Mottaleb [18] introduced a method based on probability distribution function (PDF) of face color and the mean shift estimator to track human face in the case of complete occlusion. Tu et.al. [19] introduced a camera mouse driven by a 3-D model based on face tracking techniques. Mouse cursor was navigated by head orientation and translation. Then, mouth movement was detected to control three mouse events (i.e., right, middle and left clicks). Nanda and Fujimura [20] implemented two methods for head pose estimation including principal components analysis (PCA) and artificial neural networks (ANNs). Corcoran et.al. [21] used a method based on active appearance model (AAM) to take into account head pose variation for face tracking. Dornaika and Sappa [22] suggested an appearance-based 3-D face tracking method using concepts of online appearance models (OAMs) and image registration. Viola and Jones [23, 24] presented a multiple face detection method based on Haar-like features and the AdaBoost learning approach in static images. Wang and Lee [29] used trilateral and quadrilateral finite element meshes to model the image intensity function for object tracking. They utilized a gradient based algorithm to attract mesh nodes toward image features by minimization of interpolation error in reconstruction of image from its nodal values and matching errors between the consecutive image frames. Molloy and Whelan [30] introduced an active mesh system for motion tracking of rigid objects to reduce main problems in active contour models (i.e., snake initialization). Behrad and Motamedi [31] proposed a method based on edge features extraction and matching as well as Kalman filter to detect and track mostly rigid objects like vehicles. The method, calculates the camera motion model using planar affine transformations. Jamasbi et.al. [32] used an active contour model for tracking vehicles. In order to deal with the 3-D motions or the aspect change problem, they calculated the motion model beyond the target area to reduce the contour tracking error (TE) in video.

Although these methods have some advantages, there are some disadvantages which restrict their applications in different FDAT domains. For instance, most of them have difficulties in finding the initial position and the size of the face region automatically, they are also sensitive to 2-D and 3-D head rotations [23-25]. 3-D face tracking algorithms employ multiple cameras which need to be calibrated before tracking [17]. Deformable contours are not suitable for face localization and tracking in cluttered backgrounds, long video sequences, and fully automatic applications [15]. Face tracking methods based on feature extraction and matching [33, 34]

مجلهٔ علمی- پژوهشی
«مهندسی برق مدرس»
دورهٔ دهم، شمارهٔ ۱، بهار ۱۳۸۹

Automatic Human Face Detection and Tracking using Cascaded Classifiers and Deformable Meshes

Ali Taimori[1], Alireza Behrad[2], Hassan Ghassemian[3*]
1- M.Sc. Student of Electrical Eng., Tech. and Eng. Faculty, Shahed Univ.
2- Assist. Prof. of Dept. of Electrical Eng., Tech. and Eng. Faculty, Shahed Univ.
3- Prof. of School of Electrical and Computer Eng., Tarbiat Modares Univ.

***P. O. Box 14115-143, Tehran, Iran**
ghassemi@modares.ac.ir

(Received: December 2008, Accepted: November 2009)

Abstract

In this paper, we propose a novel method for fully automatic detection and tracking of human heads and faces in video sequences. The proposed algorithm consists of two modules: a face detection module and a face tracking module. The Detection module, detects the face region and approximates it with an ellipse at the first frame using a modified version of AdaBoost cascaded classifier. The detection module is capable of considering the 2-D head pose rotation. The tracking module utiliyes a combination of deformable mesh energy minimization and feature matching approaches. In order to track a face, features are extracted in the face region to tessellate the human face with triangular unstructured meshes. For tracking a mesh, it is necessary to define mesh energies including internal and external energies. We have used new energy definitions for both the internal and the external energies which can accurately track rigid and non-rigid motions of a face and facial features at subsequent frames. We tested the proposed method with different video samples like cluttered backgrounds, partial illumination variations, put on glasses, and 2-D and/or 3-D rotating and translating heads. The experimental results showed that the algorithm is rotation insensitive and has high accuracy, stability and also has convergence for face detection and tracking.

Keywords: Face Detection and Tracking, Head Pose Estimation, AdaBoost Learning, Face Features, Feature Matching, Deformable Meshes.

1- Introduction

Recently, face detection and tracking (FDAT) in static images and video sequences has been found various applications such as human-computer interface systems [1], video surveillance systems like detection and monitoring of driver fatigue [2], video conferencing [3], robot navigation [4], smart cameras [5], computer games [6], biometric systems based on face and iris recognition [7, 8], virtual reality [9], head and face gesture recognition [10], and aiding handicapped people [11]. Therefore, with this wide range of applications, FDAT remains as a principal research topic in intelligent systems. The focus of the present paper is to propose new techniques for FDAT which are able to cope with some problems.

1-1 Related Work

Considering various face detection and tracking applications [1-11], different algorithms have been developed for face detection and tracking. Kim et al. [12]